

# System Design

PAGE

DATE

10

Good Data Model characteristics :-

- \* Model simplicity

It should describe only the entity that is intended to describe.

- \* Model non-redundancy

It should not include duplicated data (attributes) in the level of the one entity.

- \* Model flexibility and adaptability for future need

It should fit for future use in both horizontal and vertically, without redesigning the model.

Course reg. #	Course reg. date	Student #	Student name	Course 1 #	Course 1 name	Course 2 #	Course 2 name	Course 3 #

Course reg. #	Course reg. date	Student #

Student #	Student name	Student reg. #	Student interest	Student qualification



Student #	Course #	Course name	Course prerequisite	...

## Integrity Constraints :-

\* They are the rules that govern changing and deleting records and help keep the data in the database accurate.

\* Integrity Constraints is enforced using a set of integrity rules.

\* Data integrity rules provides necessary Controls upon the database to insure data integrity

\* There are three types of data integrity rules, they are:-

- Key integrity

- Domain integrity

- Referencial integrity

\* The integrity rules should be designed and implemented if it is not yet an inherent feature in used database

Package.

\* Key integrity :-

- Every table should have primary key (may be concatenated)

- Primary key should not be repeated in two records in the same table

- Primary key should not be assigned a null value

\* Domain integrity :-

- Appropriate Control should be assigned to insure



that no one attribute has taken on a value that is outside the allowed domain for that attribute values.  
Ex: for student grade, it should take values between 1 and 5

So, the values greater than 5 should be rejected.

### \* Referential integrity:

- the relational DB architecture implements a relationship between tables via foreign keys.
- the use of foreign keys increase the flexibility of any database, but it also increase the risk of referential integrity errors.
- Referential integrity error exists when a foreign key value in one table has no matching primary key in the related table.
- When deleting a record from a table, all the related records in other tables should be deleted also.
- Referential integrity errors can be prevented by applying one of following ways:-
  - Any record in the ~~record~~<sup>table</sup> may be deleted without regard to any other table, which may lead to accuracy problems.
  - Any record deleted in one table must be automatically followed by the deletion of all matched records in the related tables
  - Disallowing deleting any record in a table, unless all related ~~tables~~ records in all other tables



have been deleted.

→ A deletion of a record in a table must be automatically followed by setting any matched keys in all related table to the null value. By setting the foreign key to null, you are acknowledging that the record does not point back to a corresponding master record.

### Program Design :-

- \* Computerized information system may consist of one or more applications
- \* Each application may consist of one or more programs linked together by some sort of interface.
- \* Generally, information system should start by the log in screens, to allow user entering user name and password.
- \* Log in screen enforces the access rights rules to prohibit any unauthorized access to the system
- \* Each program should be designed, then described, by stating the following :-
  - Program name
  - Programming technology employed
  - The modules constituting this program with their names, interfaces between individual modules, input for each module, output generated by each module, processing done inside each module, and tests



related to each module.

→ Name and description of the related database.

\* Various types of tests should be described, they can be :-

→ Module verification tests

→ Integration tests

→ Functional tests

→ Performance tests

→ Installation tests

→ Acceptance tests

\* For each Type of tests the following documentation is prepared :-

→ List of tests (test code, name, brief description)

→ Test description (test code, name, description, execution steps, and expected result for each test)

→ Test report to register the status of executing the test.

\* Information system is supported by data base, which should be always up-to-date.

\* To keep database always up-to-date, it should be updated continuously transactions

\* Database that are not up-to-date is obsolete.

\* Database transactions are designed to update various attributes in database.

\* Transaction can be processed in one of two modes :-

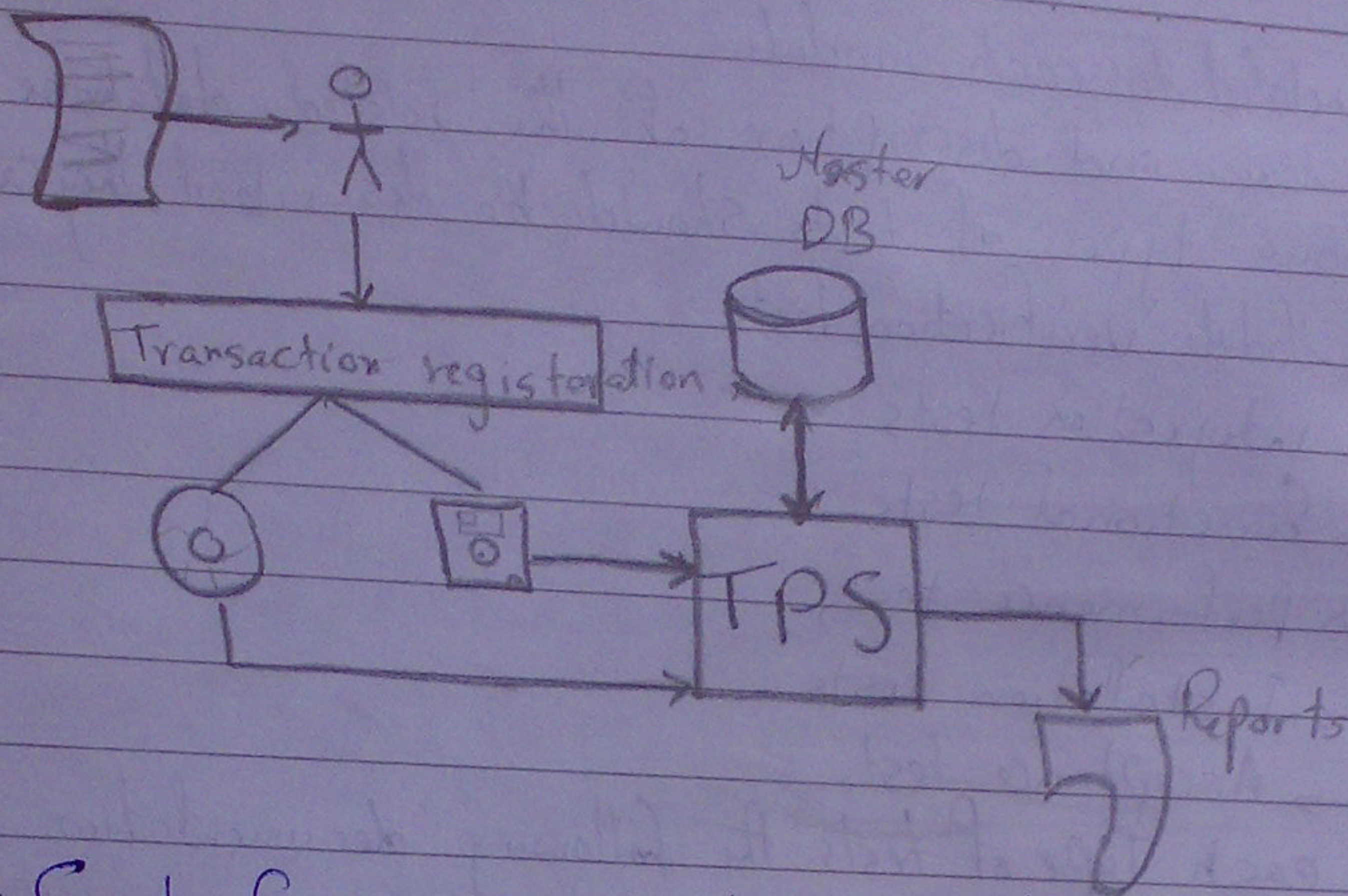
→ Batch transaction processing



# Transaction

PAGE

DATE



\* Formate for each transaction should be stated

\* Transaction id number Format Can be :-

→ \* → Transaction type

→ \* → Transaction data

→ \* → Transaction id number

\* Transaction data depends on the transaction type :-

Ex :-

Customer #
Customer name
Customer address
Customer data

Needing to update :

\* Customer address

\* % data

\* = invoice data

Customer #
Invoice #
Invoice date
Invoice data

\* Invoice item qty

\* Invoice item price

Invoice #
Item #
Item data
Item Price

\* This means we have 4 transaction types :-

\* Formate of each transaction Can be :-

Transaction Type	Trans. Id #	Trans. data
------------------	-------------	-------------

1 → Customer # → Cust add

2 → Customer # → % data

3 → Customer/Invoice # → inv. data

4 → Invoice/Item → Item qty

5 → Invoice/Item → Item price